Idaho Professional-Technical Education

	Programming & Software Development Standards Criticality Survey Results	Nice to Know	Need to Know	Critical to Know	Rating Average	Response Count
CONTENT STANDARD 1:	**Demonstrate critical thinking and problem-solving skills as they apply to programming**					
1.1	Apply basic programming principles.	0	4	25	2.86	29
1.2	Describe and differentiate procedural and object-oriented programming.	4	14	11	2.24	29
1.3	Apply the features of object-oriented programming languages	3	9	17	2.48	29
1.4	Write a program that produces output.	2	6	21	2.66	29
1.5	Select identifiers to use within programs.	7	10	11	2.14	28
1.6	Improve programs by adding comments.	2	12	15	2.45	29
1.7	Write and run a program.	2	4	23	2.72	29
CONTENT STANDARD 2:	**Demonstrate ability to use variables, data types, and string manipulation to solve computer problems programmatically**					
2.1	Demonstrate the process of declaring variables.	0	7	22	2.76	29
2.2	Display variable values.	0	12	17	2.59	29
2.3	Apply integral data types.	1	12	16	2.52	29
2.4	Apply floating-point data types	4	10	15	2.38	29
2.5	Apply arithmetic operators.	0	13	16	2.55	29
2.6	Apply Boolean data type.	0	10	19	2.66	29
2.7	Apply numeric type conversion.	2	11	16	2.48	29
2.8	Apply char data type.	3	10	16	2.45	29
2.9	Apply string data type.	1	11	17	2.55	29
2.10	Define named constants and enumerations.	4	12	13	2.31	29
CONTENT STANDARD 3:	**Demonstrate effective use of selection structures to add logic to programs**					
3.1	Demonstrate logic-planning tools and decision-making	3	10	16	2.45	29
3.2	Make decision using the if statement.	0	7	22	2.76	29
3.3	Make decisions using the if-else statement.	0	9	20	2.69	29
3.4	Apply compound expressions in if statements.	2	12	15	2.45	29
3.5	Make decisions using the switch statement.	6	7	16	2.34	29
3.6	Apply the conditional operator.	2	7	19	2.61	28
3.7	Apply the NOT operator.	4	6	19	2.52	29
3.8	Describe how to avoid common errors when making decisions, and apply problem-solving skills in context.	0	11	18	2.62	29
CONTENT STANDARD 4:	**Demonstrate ability to test, debug and validate programming applications**					
4.10	Locate a logic error by stepping through the code	1	4	24	2.79	29
4.2	Locate logic errors using breakpoints.	2	7	20	2.62	29
4.3	Fix syntax and logic errors.	1	6	22	2.72	29
4.4	Select appropriate test data for an application.	4	11	14	2.34	29
CONTENT STANDARD 5:	**Differentiate between the various types of repetition structures and use each repetition structure appropriately in program development.**					
5.1	Apply the loop structure.	0	7	20	2.74	27
5.2	Create loops using the while statement.	1	11	16	2.54	28
5.3	Create loops using the for statement.	1	11	16	2.54	28
5.4	Create loops using the do statement.	3	12	12	2.33	27
5.5	Apply nested loops.	2	13	13	2.39	28
5.6	Apply accumulators.	5	12	11	2.21	28
5.7	Understand and describe how to improve loop performance	1	19	8	2.25	28
CONTENT STANDARD 6:	**Use methods to increase functionality and to modularize programs**					
6.1	Describe methods and implementation hiding.	6	10	12	2.21	28
6.2	Write methods with no parameters and no return value.	8	6	14	2.21	28
6.3	Write methods that require a single argument.	2	11	15	2.46	28
6.4	Write methods that require multiple arguments.	5	7	16	2.39	28
6.5	Write a method that returns a value.	1	9	18	2.61	28
6.6	Pass an array to a method.	4	12	12	2.29	28
6.7	Overload methods.	7	11	10	2.11	28
6.8	Demonstrate how to avoid methods.	13	10	4	1.67	27
6.9	Apply optional parameters.	8	13	7	1.96	28
CONTENT STANDARD 7:	**Demonstrate understanding of arrays and structure and apply concepts in program development**					
7.1	Declare an array and assign values to array elements.	2	9	16	2.52	27
7.2	Access array elements.	1	9	17	2.59	27
7.3	Search an array using a loop.	2	10	14	2.46	26
7.4	Apply multidimensional arrays.	7	10	9	2.08	26
CONTENT STANDARD 8:	**Demonstrate understanding of object-oriented programming concepts.**					
8.1	Describe and apply class concepts.	2	4	21	2.7	27
8.2	Create classes from which objects can be instantiated.	1	6	20	2.7	27
8.3	Create objects.	1	5	20	2.73	26
8.4	Create properties, including auto-implemented properties.	4	11	13	2.32	28

Idaho Professional-Technical Education

Programming & Software Development Standards Criticality Survey Results		Nice to Know	Need to Know	Critical to Know	Rating Average	Response Count
8.5	Use public fields and private methods.	4	10	14	2.36	28
8.6	Define the "this" reference.	4	11	13	2.32	28
8.7	Write constructors	4	9	15	2.39	28
8.8	Use object initializers	4	13	11	2.25	28
8.9	Overload operators	12	7	9	1.89	28
8.10	Declare an array of objects	4	15	9	2.18	28
8.11	Use sorting methods with an array of objects	6	12	10	2.14	28
8.12	Write destructors	9	12	7	1.93	28
8.13	Describe and demonstrate inheritance	4	9	15	2.39	28
8.14	Extend classes	4	14	10	2.21	28
8.15	Override base class methods	6	12	10	2.14	28
8.16	Describe how a derived class object "is an" instance of the base class	5	15	8	2.11	28
8.17	Define the object class	6	14	8	2.07	28
8.18	Use base class constructors	8	12	8	2	28
8.19	Create abstract classes	11	10	7	1.86	28
8.20	Create use interfaces	8	9	11	2.11	28
8.21	Apply extension methods	12	9	7	1.82	28
8.22	Describe the benefits of inheritance	3	12	13	2.36	28
8.23	Recognize inheritance in GUI applications	10	10	8	1.93	28
CONTENT STANDARD 9: Demonstrate exception-handling in program development						
9.1	Compare and demonstrate traditional and object-oriented error-handling methods.	2	13	12	2.37	27
9.2	Cast data types.	6	11	10	2.15	27
9.3	Catch multiple exceptions.	6	14	7	2.04	27
9.4	Apply the finally block.	7	11	9	2.07	27
9.5	Handle exceptions thrown from outside methods.	5	14	8	2.11	27
9.6	Trace exceptions through the call stack.	4	13	10	2.22	27
9.7	Create exception classes	10	11	6	1.85	27
9.8	Re-throw exceptions	8	13	6	1.93	27
CONTENT STANDARD 10: Use event handlers in programs						
10.1	Define and apply event handling.	4	12	12	2.29	28
10.2	Define and describe delegates.	6	18	3	1.89	27
10.3	Declare own events and handlers.	6	17	5	1.96	28
10.4	Use built-in event handlers.	2	17	9	2.25	28
10.5	Handle control component events.	10	14	4	1.79	28
10.6	Handle mouse and keyboard events.	7	15	6	1.96	28
10.7	Manage multiple controls	9	14	5	1.86	28
10.8	Explain how to find more information on controls and events	6	12	10	2.14	28
CONTENT STANDARD 11: Apply concepts and principles of systems planning and development						
11.1	Describe the information systems development life cycle (SDLC).	9	9	10	2.04	28
11.2	Discuss how to evaluate off-the-shelf software.	17	4	7	1.64	28
11.3	Explain reuse and its role in software development.	3	11	14	2.39	28
11.4	Describe the skills required to be an effective project manager.	18	7	3	1.46	28
11.5	List and describe the skill and activities of a project manager during project initiation, planning, execution, and closedown.	20	5	3	1.39	28
11.6	Describe the steps for identifying and selecting projects and initiating and planning projects	18	6	4	1.5	28
11.7	Explain the need for and contents of a project scope statement	12	8	8	1.86	28
11.8	Compare various methods for assessing project feasibility	16	8	4	1.57	28
CONTENT STANDARD 12: Demonstrate competency with systems analysis tools and concepts						
12.1	Compare options for designing and conducting interviews to determine system requirements.	13	12	3	1.64	28
12.2	Develop a plan for conducting an interview to determine system requirements.	17	8	3	1.5	28
12.3	Explain the advantages and pitfalls of observing workers and analyzing business documents to determine system requirements.	16	10	2	1.5	28
12.4	Plan a joint application design session.	16	10	2	1.5	28
12.5	Use prototyping during requirements determination	14	8	6	1.71	28
12.6	Select appropriate methods to elicit system requirements.	13	12	3	1.64	28
12.7	Describe how requirements determination techniques apply to development of Internet applications.	17	8	3	1.5	28
12.8	Demonstrate the logical modeling of processes through studying examples of data-flow diagrams, pseudo code, and flowcharts.	8	11	9	2.04	28
CONTENT STANDARD 13: Demonstrate knowledge of application design principles						

Idaho Professional-Technical Education

Programming & Software Development Standards Criticality Survey Results		Nice to Know	Need to Know	Critical to Know	Rating Average	Response Count
13.1	Explain the process of designing interfaces and dialogues and the deliverables for their creation.	8	16	4	1.86	28
13.2	Apply the general guidelines for interface design, including guidelines for layout design, structuring data-entry fields, providing feedback, and system help.	11	14	3	1.71	28
13.3	Concisely define each of the following key database design terms: relation, primary key, functional dependency, foreign key, referential integrity, field, data type, null value, denormalization, file organization, index, and secondary key.	3	12	12	2.33	27
13.4	Explain the role of designing databases in the analysis and design of an information system.	3	16	9	2.21	28
13.5	Transform an entity-relation (E-R) diagram into an equivalent set of well-structured (normalized) relations.	10	13	5	1.82	28
13.6	Merge normalized relations from separate user views into a consolidated set of well-structured relations.	15	8	5	1.64	28
13.7	Choose storage formats for fields in database tables.	6	13	9	2.11	28
13.8	Translate well-structured relations into efficient database tables.	5	14	9	2.14	28
13.9	Explain when to use different types of file organizations to store computer files.	14	11	3	1.61	28
13.10	Describe the purpose indexes and the important considerations in selecting attributes to be indexed.	5	15	8	2.11	28
CONTENT STANDARD 14: Demonstrate knowledge of application implementation and identify the need for ongoing application support						
14.1	Describe the process of coding, testing, and converting an organizational information system.	11	11	6	1.82	28
14.2	Outline the deliverables and outcomes of an organizational information system.	16	8	4	1.57	28
14.3	List the deliverables for documenting the system and for training and supporting users.	13	13	2	1.61	28
14.4	Compare the many modes available for organizational information system training, including self-training and electronic performance support systems.	18	9	1	1.39	28
14.5	Discuss the issues of providing suppport for end users.	11	13	4	1.75	28
14.6	Explain why application implementation sometimes fails.	14	9	5	1.68	28
14.7	Describe several factors that influence the cost of maintaining an application.	10	13	5	1.82	28